View Posts
Add New Post
Search Posts

Change Forum
Tutorials
Features
Archives
Dairy Store
Join the Cow
Edit Profile
About Us
Contact Us

Simulating Fluid Dynamics

using the BhodiNUT Proximal Shader in Cinema 4D XL version 7

Page TWO: *go to page one*

Step Four:

Now we need to create something to drop into the water. For this I created a sphere with a radius of 25.

Here's where I'll be leaving much of the step by step instruction and giving more of an overview, otherwise this tutorial would turn into a book, and that just isn't necessary.

Animate the sphere so that it 'drops' through the water surface by recording keyframes. I used keyframes at frame 0 and 45 with the sphere passing through the surface of the water at about frame 15.

This is where we begin to use proximal to distort our surface. Reopen the material and turn on the 'Displacement' channel. In the image field, use the pull down menu to choose the BhodiNut 'Proximal' shader, as shown in figure 4.1. At this time, it's probably a good idea to set the strength and height as shown as well.

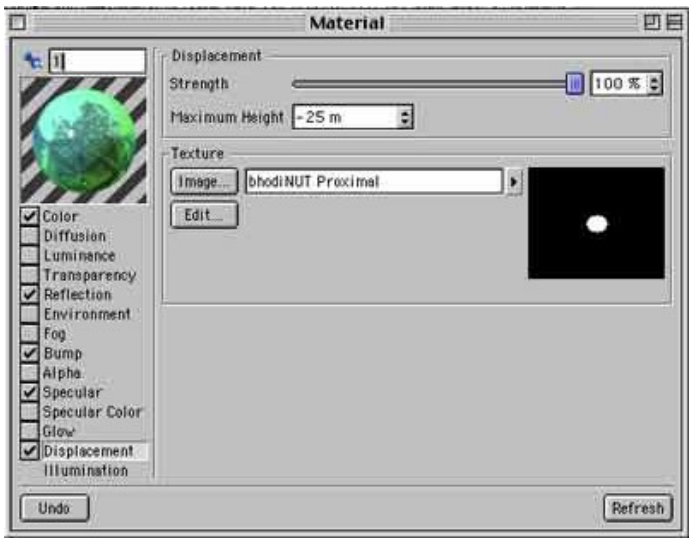

Figure 4.1

The proximal shader is wonderful! It works by using the 'proximity' of another object to influence the channel that it is assigned to. In other words, if you were to put it in the 'luminance' channel of a material applied to a cube, then assigned the proximal to be influenced by a sphere, when the sphere was moved away from the cube, the luminance value would be zero. However, as you moved the sphere closer to the cube, the luminance channel would begin to increase in strength, eventually reaching maximum value, then decreasing again as the sphere moves away.

We are going to use the Proximal shader in the displacement channel so that when our target object comes near the water surface, the water surface is distorted. To see how it works, select

the displacement channel, then click on the 'edit' button to open the dialog. When the dialog opens enter the name of your 'sphere' object in the 'search' field. Set your other settings as shown in figure 4.2. Click on 'OK' and then render a 'full rendered' preview movie to see the effect.

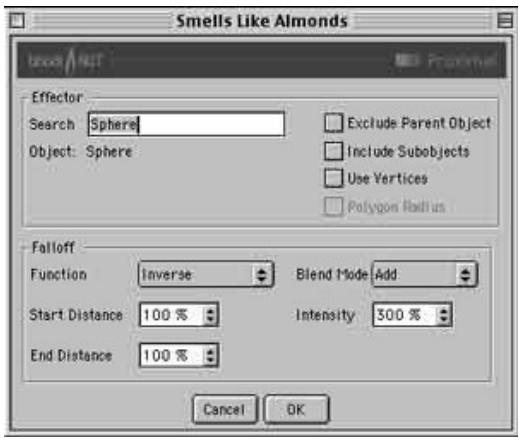

Figure 4.2

As you can see the water is distorted as if causing a splash when the sphere moves through the 'water'. However, this is very limited in influence as compared to the effect we are trying to achieve. We want the sphere to create a ripple that radiates out, strikes the walls of the container, and then bounces back in towards the point of impact. We could crank up the influence of the proximal, but that would just result in a big, unnatural distortion, so we need another solution.

Step Five:

What I did was to create a 'cone' object with a radius sized so that the trailing edge of the cone was larger than the water surface. I then animated the cone to pass through the water and targeted the proximal to the cone and turned off the rendered visibility of the cone.

Create a cone with the settings as shown in figure 5.1, then make it editable. In the version Proximal that ships with Cinema V7, the shader has been made to react to individual poly vertices (as well as poly vertices of children) as opposed to just the object axis as in earlier versions, so we upped the number of height segments to create more polys moving through the water surface. Without this ability, we would gain no benefit from using the cone, as it's axis would be no different than the sphere. But since proximal now reacts to polys, the shape of the object can determine the shape of the field of influence.

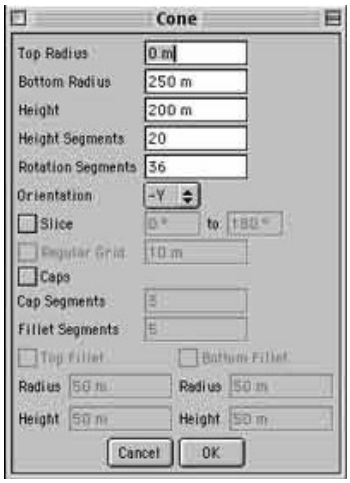

Figure 5.1

Originally, I made the cone a child of the sphere, and then just let the cone follow the sphere

through the water, but this made the ripple travel too fast. So, I animated the cone separately. You can go ahead and do this. Don't try to get it too perfect for now, as you'll more than likely be changing it's speed later.

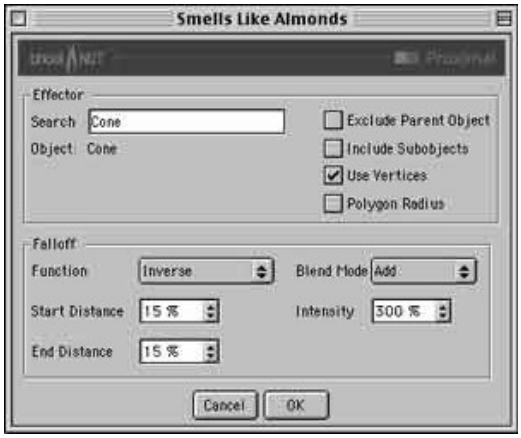

Figure 5.2

Now if we once again open the proximal dialog, and enter the settings as shown in figure 5.2, the cone should create a ripple as it passes through the water surface. Turn off the 'rendered' visibility switch of the cone, and render a frame or two as to see how the cone interacts with the water surface. See figure 5.3 and 5.4

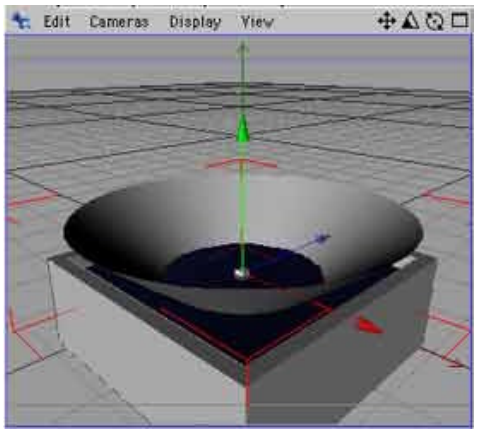

Figure 5.3

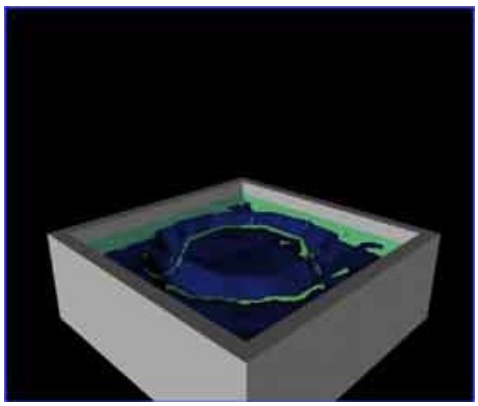

Figure 5.4

To get it to bounce off the walls, I created and grouped 4 more cones positioned so that each cone was centered over the middle of each wall of the container, then positioned higher in the y dimension so that they were just breaking the surface of the water as the original cones outer edge contacted the same point on the wall. See figure 5.5 and 5.6

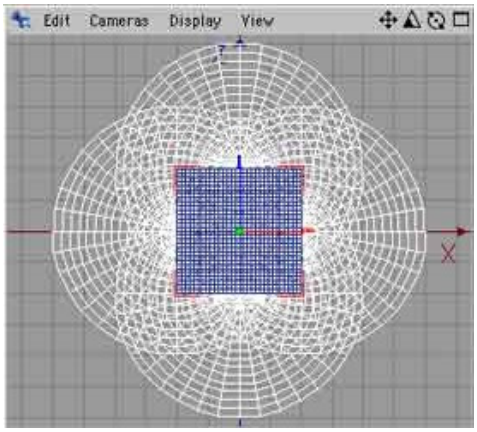

Figure 5.5

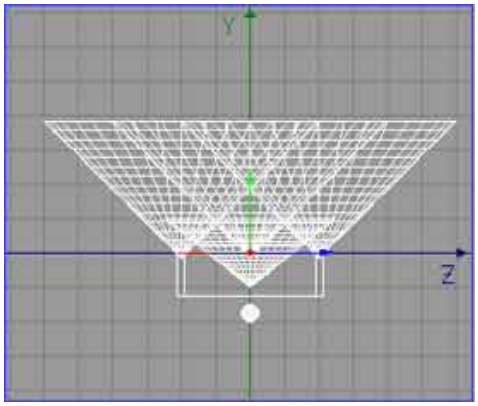

Figure 5.6

Now make the new cones a child of the original cone, and change the name of the original from 'cone' to 'cones'. We need a unique name for the main cone so that the proximal shader knows exactly which object it is assigned to. Now reopen the proximal dialog, and change the settings as shown in figure 5.7. Pay attention to the switches too. Being as the parent object is set to render invisible, the child cones will also render invisible. If you render a preview, you will see that the ripples now bounce off the walls. I wanted to add yet another layer of cones so that there was a third incidence of ripples when the second again struck the walls, to simulate real water which would become increasing complex as the ripples continued to strike and bounce off the walls. So I duplicate the group of four cones and then rotated the group 45 degrees on the 'H' axis so that it reacted to the second set of ripples when they struck the corners. See figure 5.8 and 5.9. Your hierarchy should look like that of figure 5.10.

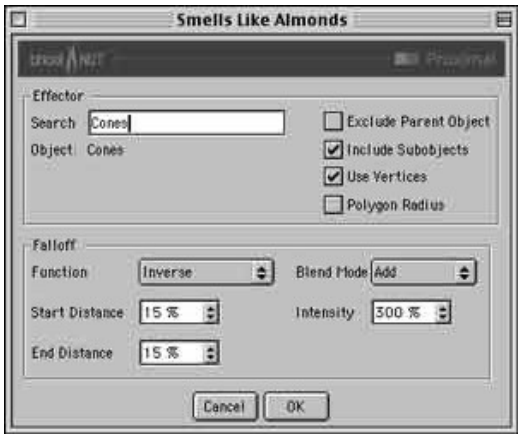

Figure 5.7

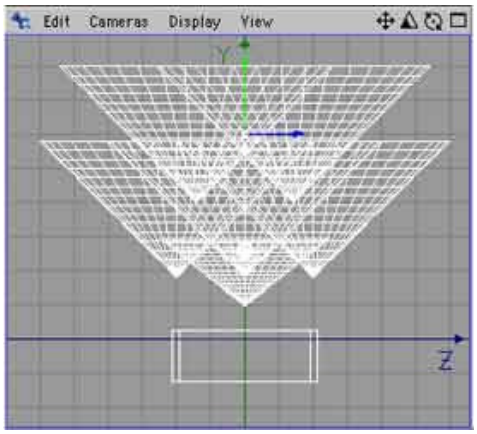

Figure 5.8

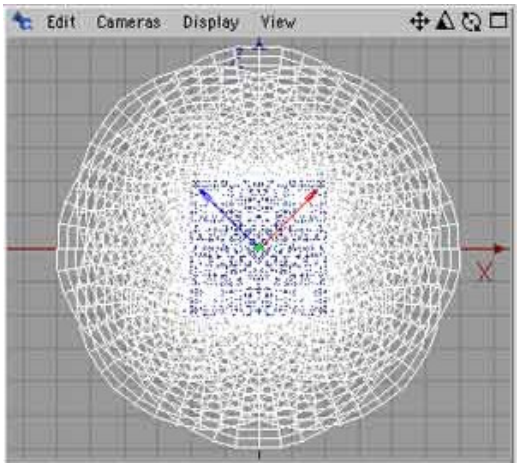

Figure 5.9

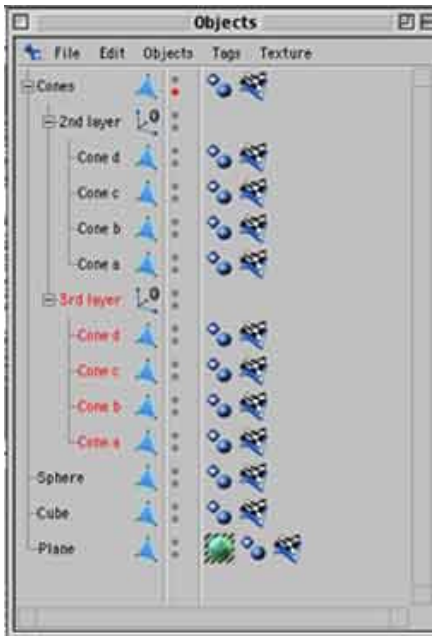

Figure 5.10

Now if you render a preview, the ripples should become more chaotic over time. This is a good time to readjust your animation so that the speed at which the cones move through the water is as realistic as you want to get. You may have to tweak the y position of your child cone groups as well.

Step Six:

Our animation is almost complete, except that the ripples remain constant in amplitude throughout the whole animation, then suddenly stop. In real life each ripple would dissipate over time, and each reflected ripple would be smaller in amplitude that the one that created it.

To simulate this effect it is only a simple matter of morphing the texture to another with a reduced displacement strength.

To do this, simply duplicate our material, and rename both materials to something meaningful. I chose '1' and '2'. Then, in the timeline, select the 'plane' object that our water material is applied to. Then create a texture track (File>New Track>Special>Effects>Texture). Control>Click on the track at the frame where the first cone penetrates the surface of the water, and enter '1' in the texture name field, click OK and a new keyframe should be created on the texture track. Then Control>Click on the texture track at the end of the animation, enter the name of the second texture in the field, click OK and a second keyframe will be created.

Now open the second material and change the displacement "Strength" to zero as shown in figure 6.0. Render out a preview and you should now see that the ripples dissipate over time.

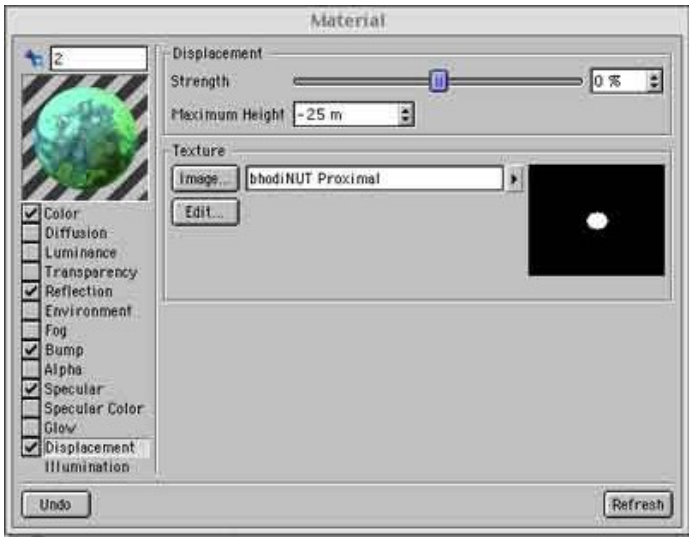

Figure 6.1

So now it's simply a matter of tweaking your animation settings to taste. As you can see there are a lot of potential applications for this technique, including raindrops, bullet holes, etc.. Hope you enjoyed/learned from this tutorial, and if you have any questions/comments be sure to drop in the cinema 4D forum and ask.

--Mark Simpson

Mark Simpson is a leader in the Cinema4D COW. Discuss this tutorial or other questions with him there.

Would you like to know more about Mark and see other articles written by him? Click here.

Please visit our forums and view other articles at CreativeCOW.net if you found this page from a direct link.

[Top of Page]